CLAIMS

What is claimed is:

1. A method of checking a data transfer for an error, the method comprising the steps of:

assuming the data transfer includes at least one aligned direct data placement (DDP) segment;

identifying a boundary of the data transfer based on an assumption that a first two bytes of a transmission control protocol (TCP) payload includes a length field of a marker with protocol data unit alignment (MPA) protocol frame; and

calculating a cyclical redundancy check (CRC) value based on the assumptions.

2. The method of claim 1, wherein the calculating step further includes calculating a TCP checksum value in parallel with calculating the CRC value.

3. The method of claim 1, wherein the identifying step includes fetching an Initial Sequence Number from a connection context and determining a remainder between the Initial Sequence Number and a data transfer sequence number.

4. The method of claim 1, wherein an MPA request/reply frame includes a correction factor including a number of bytes needed to make an initial sequence number of the data transfer word-aligned.

5. The method of claim 4, wherein the identifying step includes implementing a sequence number mod 512 process.

6. The method of claim 1, wherein the data transfer includes an MPA length field that includes an entire MPA frame including: fourteen most-significant bits (MSBs) of the MPA length field, an upper layer protocol (ULP) payload length, all MPA markers, CRC data, two least-significant bits (LSBs) of the MPA length field, and any valid bits in MPA padding.

7. The method of claim 6, further comprising the steps of:

determining whether a first word of the MPA frame equals zero;

reading the MPA length field from a next word where the first word equals zero; and

reading the MPA length field from the first word where the first word does not equal zero.

8. The method of claim 1, wherein when the data transfer does not include an MPA marker.

9. A system for checking a data transfer for an error, the system comprising:

means for identifying a boundary of the data transfer based on a first assumption that a first two bytes of a transmission control protocol (TCP) payload includes a marker with protocol data unit alignment (MPA) protocol length field; and

means for calculating a cyclical redundancy check (CRC) value based on the first assumption and a second assumption that the data transfer includes at least one aligned direct data placement (DDP) segment.

10. The system of claim 9, wherein the calculating means further includes means for calculating a TCP checksum value in parallel with calculating the CRC value.

11. The system of claim 9, wherein the identifying means includes means for fetching an Initial Sequence Number from a connection context and determining a remainder between the Initial Sequence Number and a data transfer sequence number.

12. The system of claim 9, wherein an MPA request/reply frame includes a correction factor including a number of bytes needed to make an initial sequence number of the data transfer word-aligned.

13. The system of claim 12, wherein the identifying means includes means for implementing a sequence number mod 512 process.

14. The system of claim 9, wherein the data transfer includes an MPA length field that includes an entire MPA frame including: fourteen most-significant bits (MSBs) of the MPA length field, an upper layer protocol (ULP) payload length, all MPA markers, CRC data, two least-significant bits (LSBs) of the MPA length field, and any valid bits in MPA padding.

15. The system of claim 14, further comprising:

means for determining whether a first word of the MPA frame equals zero;

means for reading the MPA length field from a next word where the first word equals zero; and

means for reading the MPA length field from the first word where the first word does not equal zero.

16. The system of claim 9, wherein when the data transfer does not include an MPA marker.

17. A computer program product comprising a computer useable medium having computer readable program code embodied therein for limiting a number of re-transmission attempts for checking a data transfer for an error, the program product comprising:

program code configured to identify a boundary of the data transfer based on a first assumption that a first two bytes of a transmission control protocol (TCP) payload includes a marker with protocol data unit alignment (MPA) protocol length field; and

program code configured to calculate a cyclical redundancy check (CRC) value based on the first assumption and a second assumption that the data transfer includes at least one aligned direct data placement (DDP) segment.

18. The program product of claim 17, wherein the calculating program code further includes program code configured to calculate a TCP checksum value in parallel with calculating the CRC value.

19. The program product of claim 17, wherein an MPA request/reply includes a correction factor including a number of bytes needed to make an initial sequence number of the data transfer word-aligned.

20. The program product of claim 19, wherein the identifying program code includes program code configured to implement a sequence number mod 512 process.